

# CSCI-5448 OOAD

## BOOKWISE - Project Part 2

**Team:** Aayush Grover, Alimulla Shaik, Shuzhe Zhang, Zhihao Wu

**Title:** BOOKWISE (An android app for exchanging and selling textbooks)

### Project Summary:

An Android App where students can post their textbooks for sale; users(students) would be able to search for a book, buy it or exchange it for a book that the other user(seller) requires. Books will be categorized for different subjects and will have information of condition, edition, picture of cover page and location. All data will be stored in a database.

### Project Requirements:

Business Requirement				
ID	Requirement	Topic Area	User	Priority
BR - 001	All users must use university email id to create profile.	Authentication	All	Critical

User Requirements				
ID	Requirement	Topic Area	User	Priority
UR - 001	As a system admin, I want to see user profile as soon as it is created so that I can keep track of all the users of the system.	User Record Maintenance	Admin	Medium
UR - 002	As a customer service representative, I want to be notified of all failures (during login and payment etc) so that I can respond to corresponding user within an hour.	System Validation	Customer Service Representative	High
UR - 003	Only seller maintains all the information about book details such as updating book availability status, rejecting reservations and deleting book details when it is sold.	Book Profile	Seller	High
UR - 004	Buyer can search for the books, he is interested in and can reserve the same if it is available.	Book Reservation	Buyer	High

---

Functional Requirements				
ID	Requirement	Topic Area	User	Priority
FR - 001	System will lists some of the recent books available in home page screen, when user logged into system.	System Display	System	Medium
FR - 002	If buyer selects any book in which he is interested, then system will notify seller of the corresponding book about the same.	Notification to Seller	System	High

Non Functional Requirements				
ID	Requirement	Topic Area	User	Priority
NFR - 001	Login functionality shall behave the same on different platforms (Linux/Windows/iOS/mobile).	Portability	System	High
NFR - 002	Restart the system in case of failures.	Reliability	System	Medium
NFR - 003	Upon providing correct credentials, user shall be directed to the home page within 10 seconds max.	Performance	System	Medium
NFR - 004	System will be open sourced.	Legal	System	Medium

### Users and Tasks:

In the system, we have two types of user: seller and buyer. Both seller and buyer should be able to register, login and logout. The buyer should be able to search for book by title, reserve book and cancel reservation. The seller should be able to post books for sale, reject buyer's reservation and delete posted book. In our android APP, we will provide various of activities such as register, search and result activities interacting with a central database to support those tasks. Use cases documents are shown below:

### Use Case List

Use Case ID	Primary Actor	Use Cases
UC-1.1.1	General User	Register
UC-1.1.2	General User	Sign In
UC-1.1.3	General User	Logout

UC-1.1.4	General User	Provide Valid E-mail Address
UC-1.1.5	General User	Create Valid Password
UC-1.1.6	General User	Provide Valid Password
UC-1.1.7	General User	Provide Contact Information
UC-1.2.1	Buyer	Search For Books
UC-1.2.2	Buyer	Reserve Books
UC-1.2.3	Buyer	Cancel Reservation
UC-1.3.1	Seller	Posting Books
UC-1.3.2	Seller	Reject Reservation
UC-1.3.3	Seller	Delete Posting

#### Use cases

<b>Use Case ID:</b>	UC-1.1.1		
<b>Use Case Name:</b>	Register		
<b>Created By:</b>	Shuzhe Zhang	<b>Last Updated By:</b>	Shuzhe Zhang
<b>Date Created:</b>	Oct 6, 2015	<b>Last Revision Date:</b>	Oct 6, 2015
<b>Actors:</b>	General User (Any User)		
<b>Description:</b>	This use case allow a user to register for a new account		
<b>Frequency of Use:</b>	Once per User		
<b>Preconditions:</b>	User does not have an account		
<b>Postconditions:</b>	User has an account		
<b>Normal Flow:</b>	<ol style="list-style-type: none"> <li>1. User visit the register page</li> <li>2. User provide valid @colorado.edu E-mail address</li> <li>3. User create valid password (at least 8 digits)</li> <li>4. Provide contact information (Mandatory/optional)</li> <li>5. User click on register button</li> </ol>		
<b>Exceptions:</b>	<ol style="list-style-type: none"> <li>1. E-mail address is invalid</li> <li>2. Password is invalid</li> <li>3. Mandatory contact information not provided</li> <li>4. Account Already exist</li> </ol>		
<b>Includes:</b>	Provide Valid Email Address Create Valid Password Provide Contact Information		
<b>Special Requirements:</b>	None		
<b>Assumptions:</b>	User has internet access		

<b>Notes and Issues:</b>	None
--------------------------	------

<b>Use Case ID:</b>	UC-1.1.2		
<b>Use Case Name:</b>	Sign In		
<b>Created By:</b>	Shuzhe Zhang	<b>Last Updated By:</b>	Shuzhe Zhang
<b>Date Created:</b>	Oct 6, 2015	<b>Last Revision Date:</b>	Oct 6, 2015
<b>Actors:</b>	General User (Any User)		
<b>Description:</b>	This use case allow a user to sign in with his/her account		
<b>Frequency of Use:</b>	Every time switch account or sign in to a new device		
<b>Preconditions:</b>	User was logged out		
<b>Postconditions:</b>	User is signed in		
<b>Normal Flow:</b>	1. User visit the sign in page 2. User provide valid @colorado.edu E-mail address 3. User provide valid password 4. User click on sign in button		
<b>Exceptions:</b>	1. E-mail address is invalid 2. Password is invalid		
<b>Includes:</b>	Provide Valid E-mail Address Provide Valid Password		
<b>Special Requirements:</b>	None		
<b>Assumptions:</b>	User has internet access		
<b>Notes and Issues:</b>	None		

<b>Use Case ID:</b>	UC-1.1.3		
<b>Use Case Name:</b>	Log out		
<b>Created By:</b>	Shuzhe Zhang	<b>Last Updated By:</b>	Shuzhe Zhang
<b>Date Created:</b>	Oct 6, 2015	<b>Last Revision Date:</b>	Oct 6, 2015
<b>Actors:</b>	General User (Any User)		
<b>Description:</b>	This use case allow a user to log out his/her account		
<b>Frequency of Use:</b>	Every time user wants to log out		
<b>Preconditions:</b>	User was signed in		
<b>Postconditions:</b>	User is logged out		

<b>Normal Flow:</b>	1. User click on log out button
<b>Exceptions:</b>	None
<b>Includes:</b>	None
<b>Special Requirements:</b>	None
<b>Assumptions:</b>	User has internet access
<b>Notes and Issues:</b>	None

<b>Use Case ID:</b>	UC-1.1.4		
<b>Use Case Name:</b>	Provide Valid E-mail Address		
<b>Created By:</b>	Shuzhe Zhang	<b>Last Updated By:</b>	Shuzhe Zhang
<b>Date Created:</b>	Oct 6, 2015	<b>Last Revision Date:</b>	Oct 6, 2015
<b>Actors:</b>	General User (Any User)		
<b>Description:</b>	This use case allow a user to provide a valid E-mail address		
<b>Frequency of Use:</b>	Every time user wants to register or sign in		
<b>Preconditions:</b>	Email address not provided		
<b>Postconditions:</b>	Email address provided		
<b>Normal Flow:</b>	1. User click on E-mail address input field 2. User type in E-mail address		
<b>Exceptions:</b>	None		
<b>Includes:</b>	None		
<b>Special Requirements:</b>	None		
<b>Assumptions:</b>	User has internet access		
<b>Notes and Issues:</b>	None		
<b>Use Case ID:</b>	UC-1.1.5		
<b>Use Case Name:</b>	Create Valid Password		
<b>Created By:</b>	Shuzhe Zhang	<b>Last Updated By:</b>	Shuzhe Zhang
<b>Date Created:</b>	Oct 6, 2015	<b>Last Revision Date:</b>	Oct 6, 2015
<b>Actors:</b>	General User (Any User)		
<b>Description:</b>	This use case allow a user to create a valid password		

<b>Frequency of Use:</b>	Every time user wants to register		
<b>Preconditions:</b>	Password not exist		
<b>Postconditions:</b>	Password created		
<b>Normal Flow:</b>	1. User click on the new password input field 2. User type in a new password		
<b>Exceptions:</b>	None		
<b>Includes:</b>	None		
<b>Special Requirements:</b>	None		
<b>Assumptions:</b>	User has internet access		
<b>Notes and Issues:</b>	None		
<b>Use Case ID:</b>	UC-1.1.6		
<b>Use Case Name:</b>	Provide Valid Password		
<b>Created By:</b>	Shuzhe Zhang	<b>Last Updated By:</b>	Shuzhe Zhang
<b>Date Created:</b>	Oct 6, 2015	<b>Last Revision Date:</b>	Oct 6, 2015
<b>Actors:</b>	General User (Any User)		
<b>Description:</b>	This use case allow a user to provide a valid password		
<b>Frequency of Use:</b>	Every time user wants to sign in		
<b>Preconditions:</b>	Password not provided		
<b>Postconditions:</b>	Password provided		
<b>Normal Flow:</b>	1. User click on the password input field 2. User type in the password		
<b>Exceptions:</b>	None		
<b>Includes:</b>	None		
<b>Special Requirements:</b>	None		
<b>Assumptions:</b>	User has internet access		
<b>Notes and Issues:</b>	None		
<b>Use Case ID:</b>	UC-1.1.7		
<b>Use Case Name:</b>	Provide Contact Information		
<b>Created By:</b>	Shuzhe Zhang	<b>Last Updated By:</b>	Shuzhe Zhang

<b>Date Created:</b>	Oct 6, 2015	<b>Last Revision Date:</b>	Oct 6, 2015
<b>Actors:</b>	General User (Any User)		
<b>Description:</b>	This use case allow a user to provide his/her contact information		
<b>Frequency of Use:</b>	Every time user wants to register		
<b>Preconditions:</b>	Contact information (mandatory/optional) not provided		
<b>Postconditions:</b>	Contact information (mandatory/optional) provided		
<b>Normal Flow:</b>	<ol style="list-style-type: none"> <li>1. User click on the contact information input field</li> <li>2. User type in the contact information</li> </ol>		
<b>Exceptions:</b>	None		
<b>Includes:</b>	None		
<b>Special Requirements:</b>	None		
<b>Assumptions:</b>	User has internet access		
<b>Notes and Issues:</b>	None		

<b>Use Case ID:</b>	UC-1.2.1		
<b>Use Case Name:</b>	Search Books		
<b>Created By:</b>	Shuzhe Zhang	<b>Last Updated By:</b>	Shuzhe Zhang
<b>Date Created:</b>	Oct 6, 2015	<b>Last Revision Date:</b>	Oct 6, 2015
<b>Actors:</b>	Buyer		
<b>Description:</b>	This use case allow a Buyer to search for book he/she wants to buy		
<b>Frequency of Use:</b>	Every time Buyer wants to buy a book		
<b>Preconditions:</b>	Buyer was on search page		
<b>Postconditions:</b>	Buyer has a list of results		
<b>Normal Flow:</b>	<ol style="list-style-type: none"> <li>1. Buyer visit the search page</li> <li>2. Buyer provide title/author/edition</li> <li>3. Buyer select sort with distance/price option</li> <li>4. Buyer click on search</li> </ol>		
<b>Exceptions:</b>	None		
<b>Extends:</b>	Sort Results With Distance Sort Results With Price		
<b>Special Requirements:</b>	None		

<b>Assumptions:</b>	User has internet access
<b>Notes and Issues:</b>	None

<b>Use Case ID:</b>	UC-1.2.2		
<b>Use Case Name:</b>	Reserve Books		
<b>Created By:</b>	Shuzhe Zhang	<b>Last Updated By:</b>	Shuzhe Zhang
<b>Date Created:</b>	Oct 6, 2015	<b>Last Revision Date:</b>	Oct 6, 2015
<b>Actors:</b>	Buyer		
<b>Description:</b>	This use case allow a Buyer to reserve a book he/she wants to buy		
<b>Frequency of Use:</b>	Every time Buyer found a book he/she wants to buy		
<b>Preconditions:</b>	Buyer searched book and has a list of results, but does not have reservation on the book		
<b>Postconditions:</b>	Buyer has reservation/on the wait list of the book		
<b>Normal Flow:</b>	1. Buyer click on the reserve button		
<b>Exceptions:</b>	None		
<b>Includes:</b>	None		
<b>Special Requirements:</b>	None		
<b>Assumptions:</b>	User has internet access		
<b>Notes and Issues:</b>	None		

<b>Use Case ID:</b>	UC-1.2.3		
<b>Use Case Name:</b>	Cancel Reservation		
<b>Created By:</b>	Shuzhe Zhang	<b>Last Updated By:</b>	Shuzhe Zhang
<b>Date Created:</b>	Oct 6, 2015	<b>Last Revision Date:</b>	Oct 6, 2015
<b>Actors:</b>	Buyer		
<b>Description:</b>	This use case allow a Buyer to cancel a reservation he/she made before		
<b>Frequency of Use:</b>	Every time Buyer decide not to buy a book		
<b>Preconditions:</b>	Buyer has a reservation on a book		
<b>Postconditions:</b>	Buyer dose not have a reservation on the book		



<b>Normal Flow:</b>	1. Buyer visit the reserved book page 2. Buyer click on the cancel reservation button
<b>Exceptions:</b>	None
<b>Includes:</b>	None
<b>Special Requirements:</b>	None
<b>Assumptions:</b>	User has internet access
<b>Notes and Issues:</b>	None

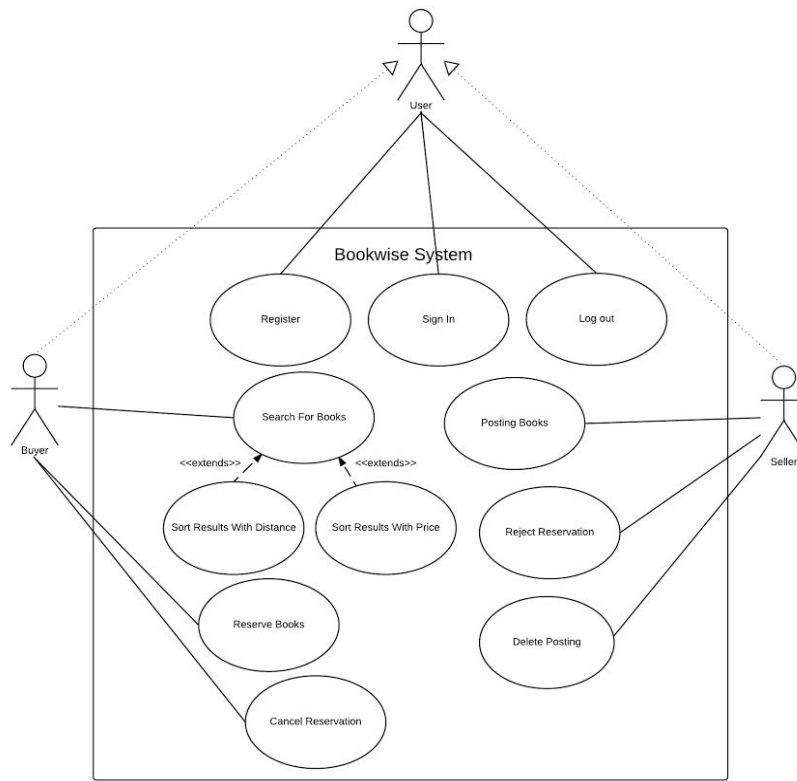
<b>Use Case ID:</b>	UC-1.3.1		
<b>Use Case Name:</b>	Post Book		
<b>Created By:</b>	Shuzhe Zhang	<b>Last Updated By:</b>	Shuzhe Zhang
<b>Date Created:</b>	Oct 6, 2015	<b>Last Revision Date:</b>	Oct 6, 2015
<b>Actors:</b>	Seller		
<b>Description:</b>	This use case allow a seller to post a book he/she wants to sell		
<b>Frequency of Use:</b>	Every time a Seller has a book to sell		
<b>Preconditions:</b>	Seller has not posted the book for sale		
<b>Postconditions:</b>	Seller has posted the book for sale		
<b>Normal Flow:</b>	1. Seller visit the book posting page 2. Seller provide book title/author/edition 3. Seller click on post button		
<b>Exceptions:</b>	None		
<b>Includes:</b>	None		
<b>Special Requirements:</b>	None		
<b>Assumptions:</b>	User has internet access		
<b>Notes and Issues:</b>	None		

<b>Use Case ID:</b>	UC-1.3.2		
<b>Use Case Name:</b>	Reject Reservation		
<b>Created By:</b>	Shuzhe Zhang	<b>Last Updated By:</b>	Shuzhe Zhang
<b>Date Created:</b>	Oct 6, 2015	<b>Last Revision Date:</b>	Oct 6, 2015

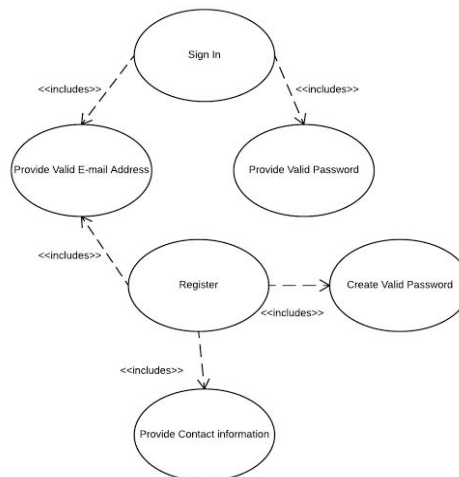
<b>Actors:</b>	Seller
<b>Description:</b>	This use case allow a seller to reject a reservation a buyer made
<b>Frequency of Use:</b>	Every time a seller decide not to sell the book to a buyer who made a reservation
<b>Preconditions:</b>	The buyer has a reservation on the book
<b>Postconditions:</b>	The buyer no longer has a reservation on the book
<b>Normal Flow:</b>	1. Seller visit the posted book page 2. Seller click on reject reservation button
<b>Exceptions:</b>	None
<b>Includes:</b>	None
<b>Special Requirements:</b>	None
<b>Assumptions:</b>	User has internet access
<b>Notes and Issues:</b>	None

<b>Use Case ID:</b>	UC-1.3.3		
<b>Use Case Name:</b>	Delete Posting		
<b>Created By:</b>	Shuzhe Zhang	<b>Last Updated By:</b>	Shuzhe Zhang
<b>Date Created:</b>	Oct 6, 2015	<b>Last Revision Date:</b>	Oct 6, 2015
<b>Actors:</b>	Seller		
<b>Description:</b>	This use case allow a seller to delete a book he/she posted		
<b>Frequency of Use:</b>	Every time a seller decide not to sell the book		
<b>Preconditions:</b>	The book was posted for sale		
<b>Postconditions:</b>	The book is no longer posted for sale		
<b>Normal Flow:</b>	1. Seller visit the posted book page 2. Seller click on delete posting button		
<b>Exceptions:</b>	None		
<b>Includes:</b>	None		
<b>Special Requirements:</b>	None		
<b>Assumptions:</b>	User has internet access		
<b>Notes and Issues:</b>	None		

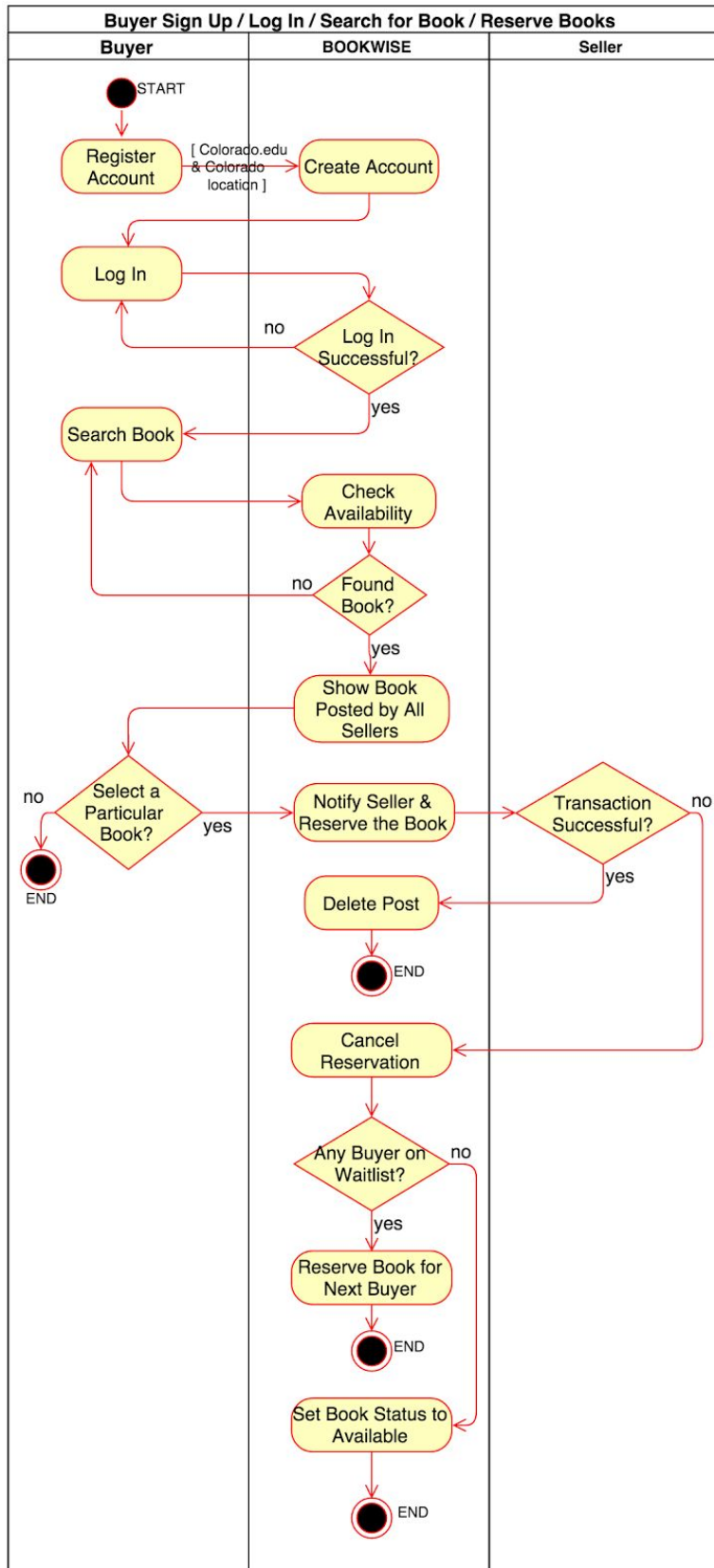
Use cases diagram:



Sub-Diagram



## Activity Diagram:



## Data Storage:

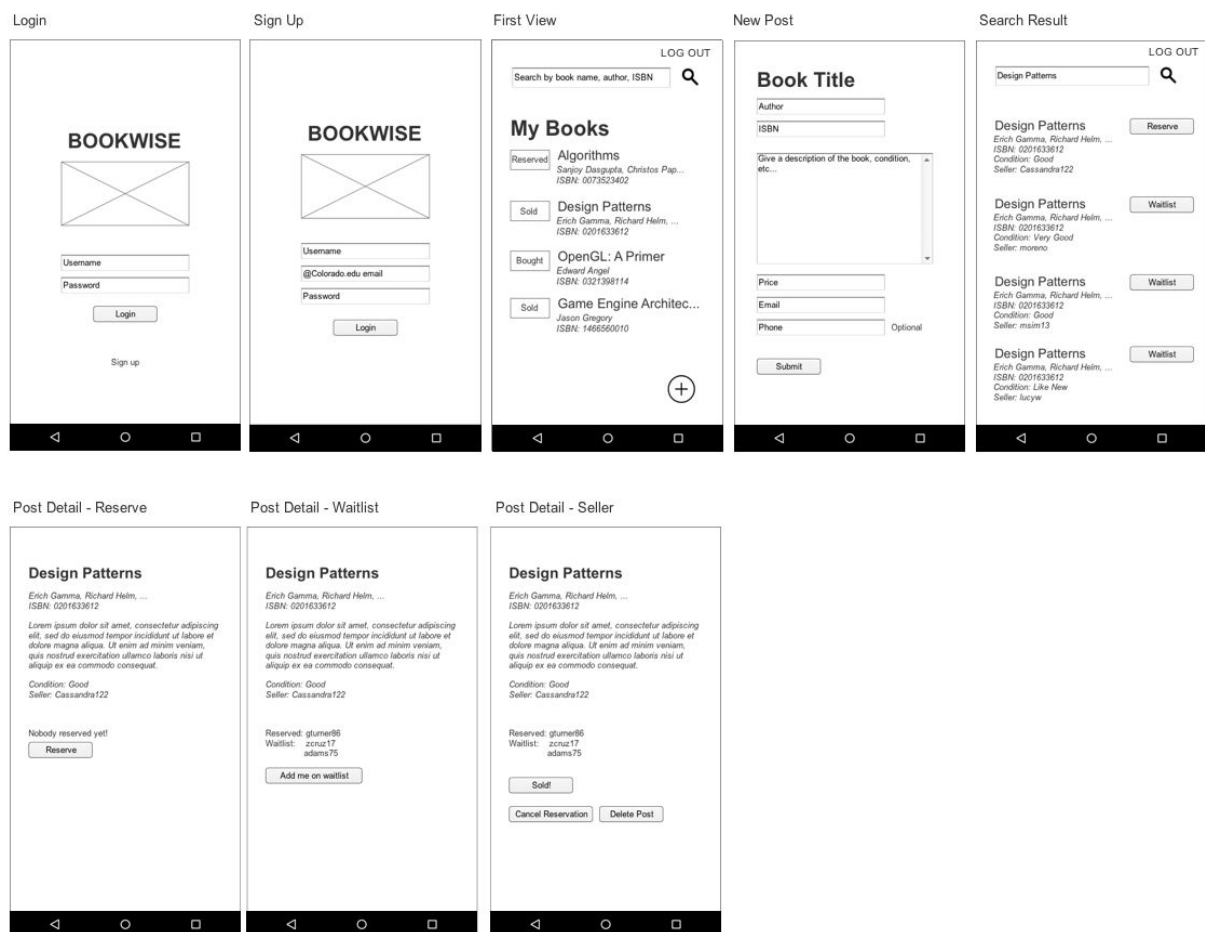
Database is created and stored by Administrator with the seller details such as book title, book author, ISBN, Description of the book, price, seller email id and contact number (optional). And the details of the buyer such as username, email id and contact number (optional).

We will use SQLite storage technology to store the data in Android. The databases are created/opened by extending SQLiteOpenHelper class using onCreate() method and the same class is used to access the data at runtime. Databases in android are stored in the path "/data/data/<app\_package\_name>/databases/<database\_name>" by default, same path can be obtained by calling Context.getDatabasePath() method in helper class constructor.

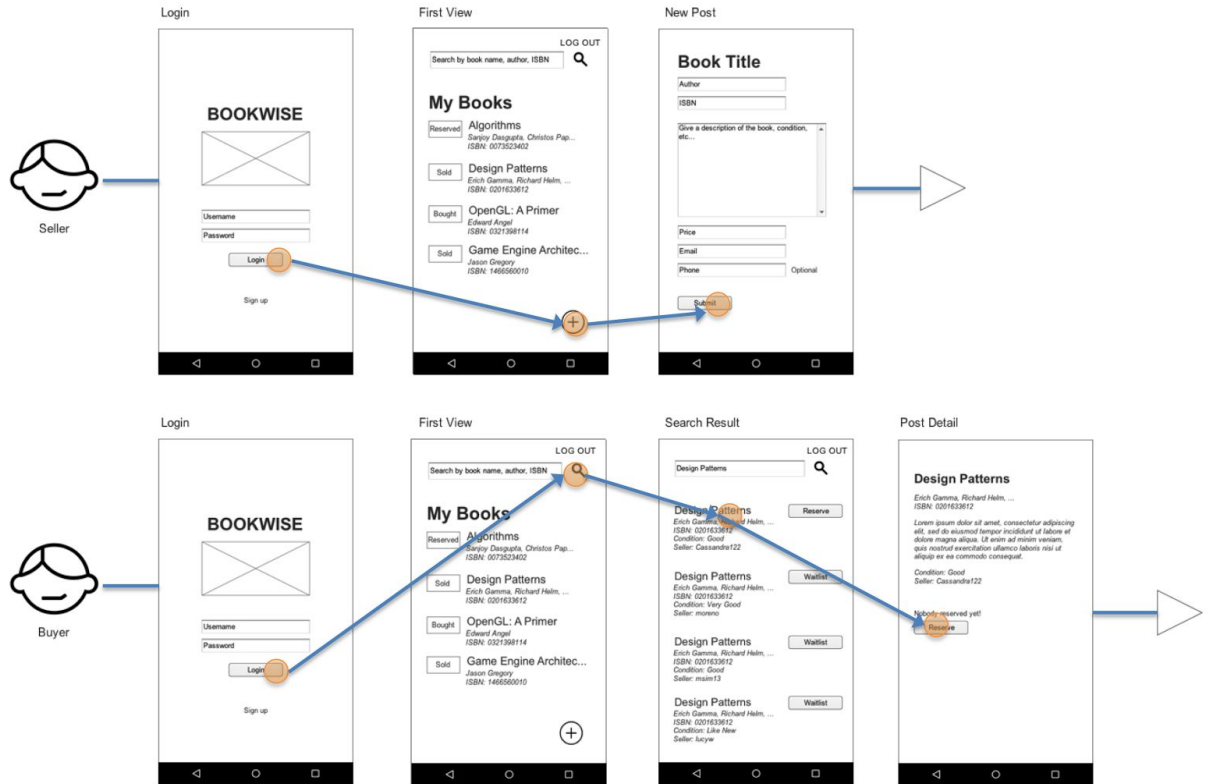
## UI Mockups:

We try to keep the UI clean and focus on the main functionalities. In Post Detail view, buyer will see different call to action buttons depending on if book is available for reservation, seller will see options like cancel reservation or delete post.

Wireframes:



The navigation of two example user interactions (add new post and search/reserve book) are shown by the user flows below:

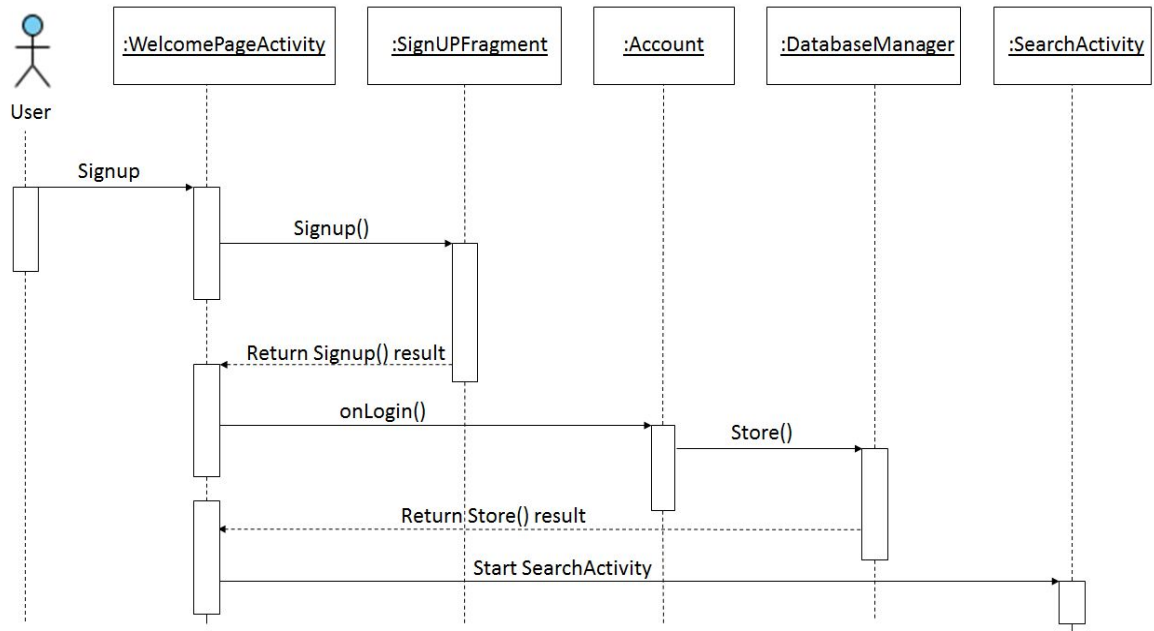


## User Interactions:

For this assignment, we are presenting the interaction of sign in, search book, reserve books.

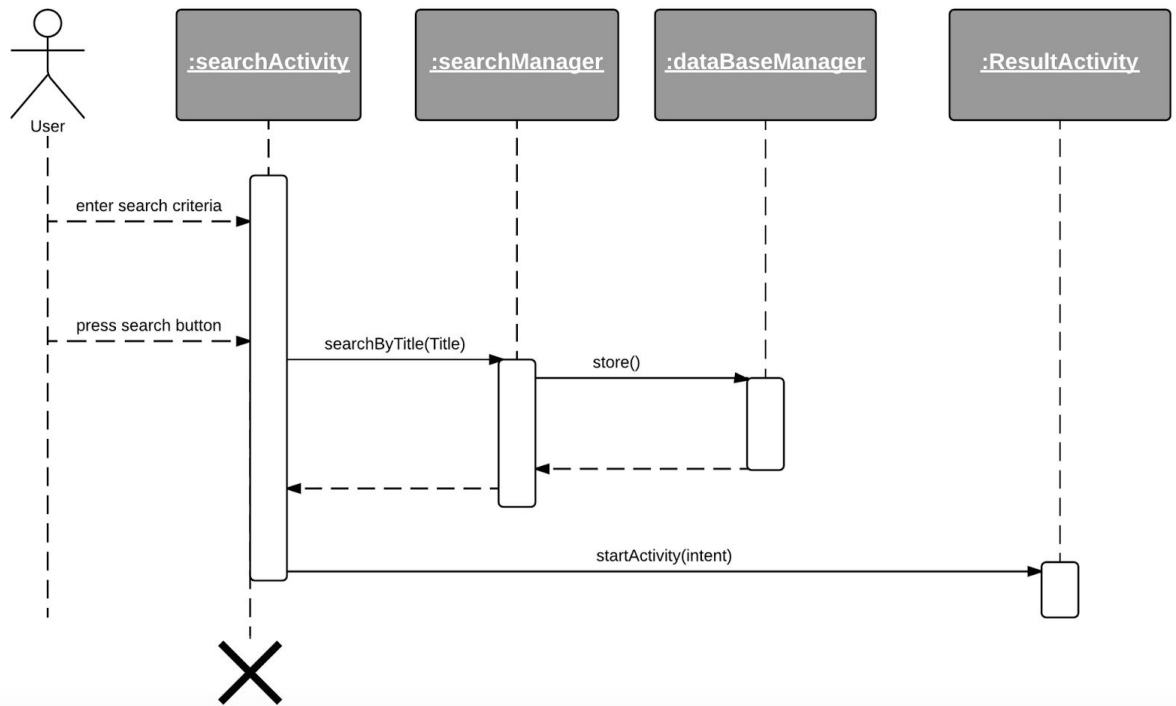
Sign In Sequence Diagram:

Sign In Sequence Diagram

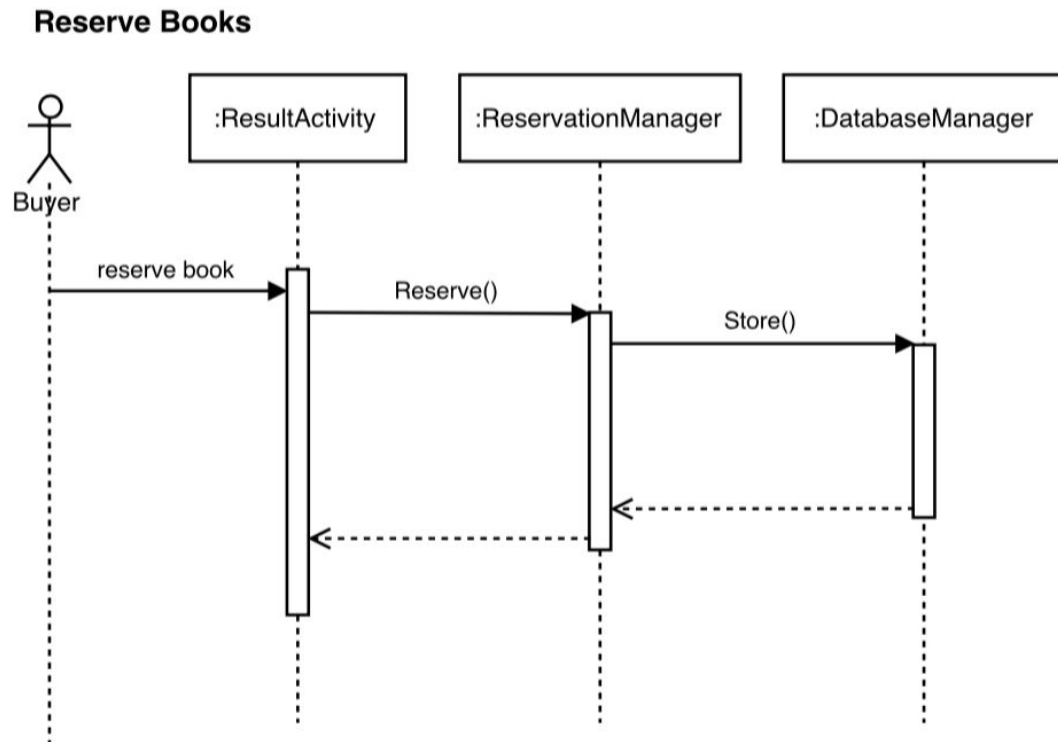


Search book sequence diagram:

Search activity sequence diagram



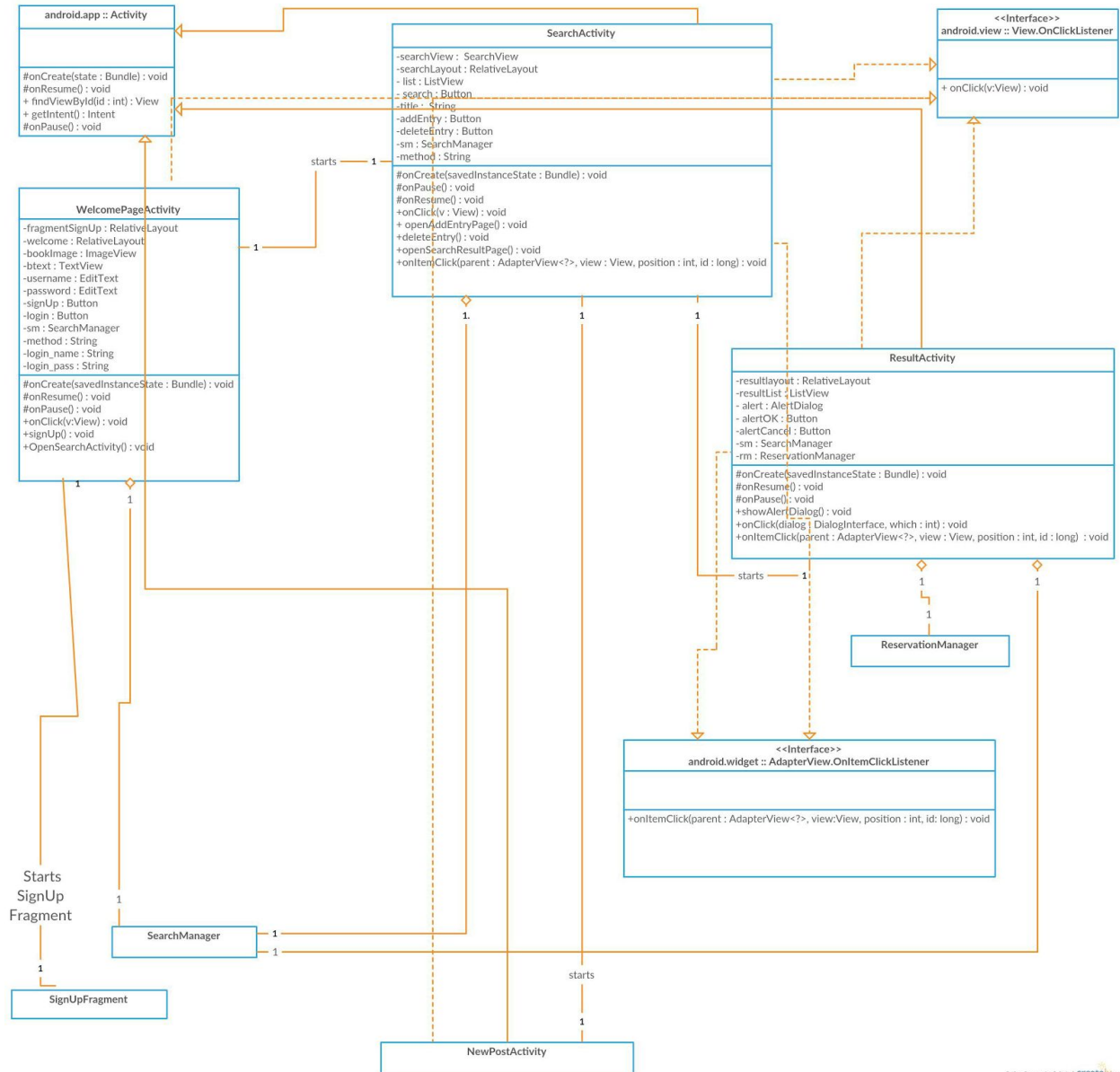
Reserve books sequence diagram:



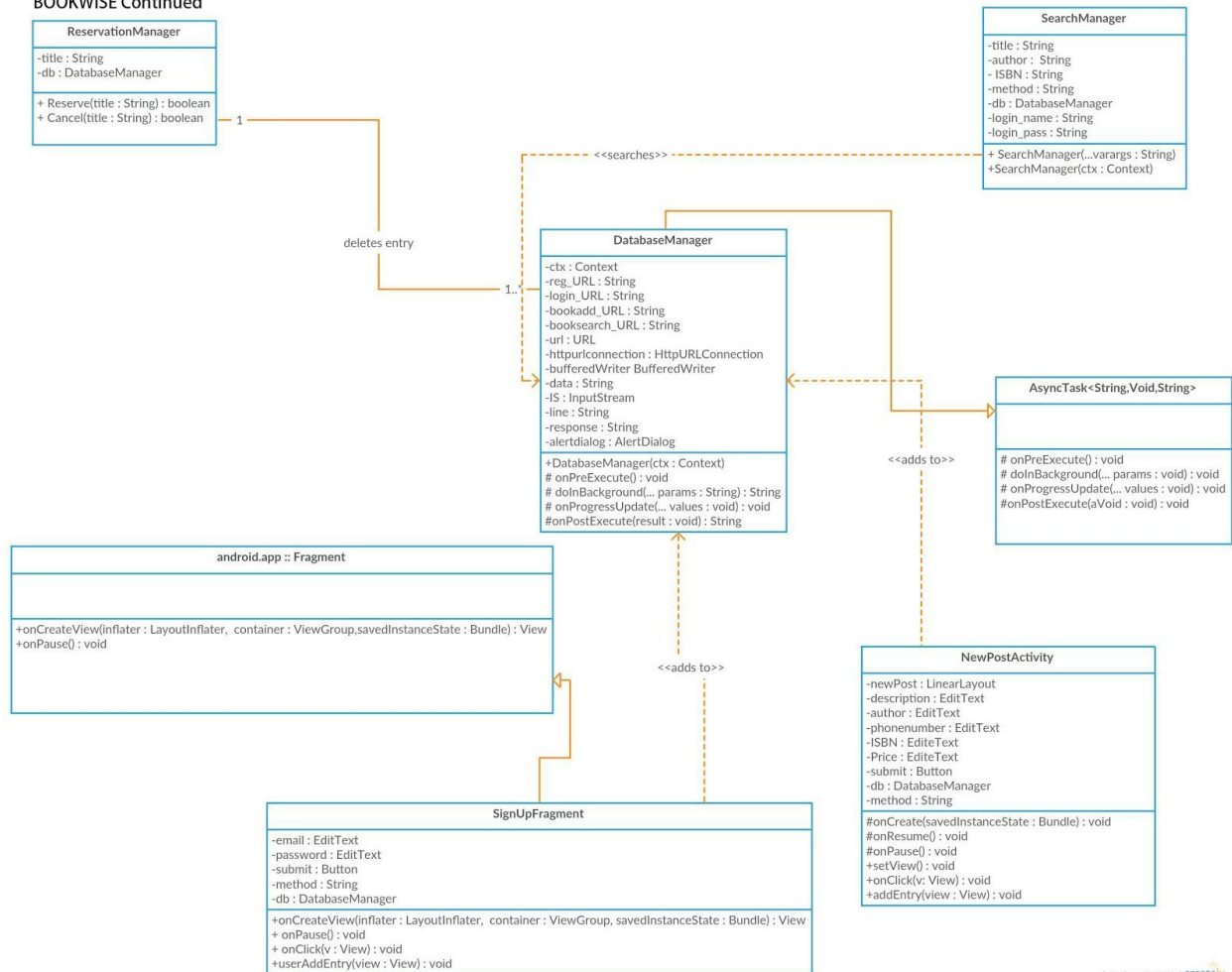


## Class Diagram:

### BOOKWISE



## BOOKWISE Continued



## Credit:

The team conducted four two-hour meetings to discuss and create every documents and diagrams. Each member of the team is responsible to document and implement different parts of the assignment:

Project Requirement:	Alimulla
Users and Tasks:	Shuzhe
Activity Diagram:	Aayush, Zhihao
Data Storage:	Alimulla
UI Mockups:	Zhihao
User Interactions:	Sign In - Alimulla
	Search Book - Shuzhe
	Reserve Books - Zhihao
Class Diagrams:	Aayush